The opinion in support of the decision being entered today was not written for publication and is not binding precedent of the Board.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte JAMES E. DIBB

Appeal No. 2006-3036
Application No. 09/879,554

ON BRIEF

MAILED
DEC 20 2006
U.S. PATENT AND TRADEMARK OFFICE
BOARD OF PATENT APPEALS
AND INTERFERENCES

Before JERRY SMITH, BARRY, and MACDONALD, Administrative Patent Judges.

JERRY SMITH, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on the appeal under 35 U.S.C. § 134 from the examiner's rejection of claims 1-14 and 16-19, which constitute all the claims pending in this application.

The disclosed invention pertains to a method and system for repairing a redundant array of disk drives. When a disk drive fails, a mirrored subsystem is created within the array that includes a temporary disk drive and the failed disk drive slot. The mirrored subsystem is substituted for the failed disk drive in the

redundancy group. Each data block of the failed disk drive is reconstructed and written to the mirrored subsystem. After a replacement disk drive is inserted into the failed disk drive slot, data is copied from the temporary disk drive to the replacement disk drive. The mirrored subsystem can be replaced by the replacement disk after the data thereon matches the data on the temporary disk drive.

Representative claim 1 is reproduced as follows:

1. A method for handling a failed disk drive in a redundancy group of disk drives in an array of disk drives, the failed disk drive located in a failed disk drive slot, comprising:

creating a mirrored subsystem within the array, the subsystem including a temporary disk drive and the failed disk drive slot; and

reconfiguring the redundancy group to consist of the disk drives of the redundancy group that have not failed and the mirrored subsystem, such that the mirrored subsystem is substituted for the failed disk drive in the redundancy group and the redundancy of the redundancy group is restored, when the failed disk drive contains redundancy data for the redundancy group [sic][1]

The examiner relies on the following reference:

Kedem	6,154,853	Nov. 28, 2000

The following rejections are on appeal before us:

1. Claims 1-14[2] stand rejected under 35 U.S.C. § 112, first paragraph, as failing to comply with the written description requirement.

---

[1] Claim 1 lacks a period. See brief, claims appendix. See also amendments filed Nov. 02, 2005 and Feb. 16, 2005. Although the amendment filed Feb. 16, 2005 deleted the period via a strikethrough, we nevertheless presume that the omission of a period is a typographical error.

[2] Although the examiner includes claim 15 in this rejection [see answer, page 3], claim 15 has been cancelled. See brief, page 21 and reply brief, page 6 (indicating cancellation of claim 15). See also answer, page 2.

2. Claims 16-19 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Kedem.

Rather than repeat the arguments of appellant or the examiner, we make reference to the briefs and the answer for the respective details thereof.

OPINION

We have carefully considered the subject matter on appeal, the rejections advanced by the examiner and the evidence of anticipation relied upon by the examiner as support for the rejections. We have, likewise, reviewed and taken into consideration, in reaching our decision, the appellant's arguments set forth in the briefs along with the examiner's rationale in support of the rejections and arguments in rebuttal set forth in the examiner's answer. Only those arguments actually made by appellant have been considered in this decision. Arguments which appellant could have made but chose not to make in the briefs have not been considered and are deemed to be waived [see 37 CFR § 41.37(c)(1)(vii)(2004)].

It is our view, after consideration of the record before us, that the specification reasonably conveys to the skilled artisan that the inventor had possession of the invention of claims 1-14 at the time the application was filed. Moreover, we conclude that the disclosure of Kedem fully meets the invention as set forth in claims 16-19. Accordingly, we affirm-in-part.

We first consider the examiner's rejection of claims 1-14 under 35 U.S.C. § 112, first paragraph as failing to comply with the written description requirement. The examiner asserts that the specification does not reasonably support the limitation in claim 1 calling for substituting the failed disk drive in the redundancy group and restoring the redundancy of the redundancy group when the failed disk drive contains redundancy data for the redundancy group [answer, page 3]. The examiner notes that the specification teaches reconstructing each "data block" of the failed drive [answer, pages 6 and 7]. But since the specification distinguishes "user data" (data blocks) from "redundancy data" (parity blocks),[3] the examiner concludes that the specification refers only to reconstructing data blocks (i.e., logical data only) – not reconstructing both data blocks and parity blocks as claimed [answer, page 8].

Appellant argues that the application as originally filed teaches using a mirrored subsystem to completely restore redundancy even when the failed drive contains redundancy data. In this regard, appellant notes that the originally-filed application expressly states that the invention reconstructs each data block of the failed drive. Accordingly, the skilled artisan would understand that such reconstruction includes all data sectors on the failed drive – sectors with logical and redundancy data [brief, pages 13 and 14; reply brief, page 2; emphasis added]. Thus, if drive A fails, the skilled artisan would understand from the originally-filed disclosure that repairing the redundant disk drive array requires

[3] Specifically, the examiner quotes the following sentence from page 5 of the specification: "Data is stored in redundancy group 7 in stripes that contain a plurality of data blocks and at least one associated error-correction block" [specification, page 5].

restoring drive A to its former condition (i.e., reconstructing all data that was formerly on drive A). Significantly, if drive A contained logical and redundancy data (or redundancy data only), reconstructing less than all data blocks (i.e., both logical and redundancy data) would not restore drive A to its former condition; consequently, the array would not be repaired. Rather, array repair under such conditions is only possible by reconstructing both logical and redundancy data [reply brief, pages 6 and 7].

We will not sustain the examiner's rejection. We agree with appellant that the skilled artisan would readily understand from the originally-filed disclosure that reconstructing the data that was formerly on the failed drive (e.g., drive A in the disclosure) necessarily requires reconstructing all data that was on the drive prior to failure. Otherwise, the array would not be repaired. Therefore, if the failed drive contains both logical and redundancy data, both the logical and redundancy data must be restored to effect repair.

Turning to claim 1, the claim calls for restoring redundancy of the redundancy group "when the failed disk drive contains redundancy data for the redundancy group." Such a limitation essentially imposes a condition: if the failed drive should contain redundancy data for the redundancy group, then redundancy is restored for that particular type of array by substituting the mirrored subsystem for the failed drive.[4]

---

[4] Reciting such a condition in the claim, however, raises an additional issue regarding the scope of the claim. Specifically, we note that the claimed condition does not recite an active step, but in effect recites alternative limitations: (1) when the failed disk drive contains redundancy data, and (2) when the failed drive does not contain redundancy data. Significantly, dependent claims 4, 5,

Although individual disk drives configured in redundant arrays do not necessarily have to contain both logical and redundancy data, the originally-filed disclosure nevertheless teaches repairing arrays with disk drives containing both types of data. For example, the disclosure expressly states that the redundancy group may be a RAID-3 or RAID-5 array [specification, pages 3-5; see also claims 4, 5, 8, and 9]. As is well known in the art, a RAID-3 array utilizes byte-level striping with a dedicated parity drive, and a RAID-5 array utilizes block-level data striping with distributed parity.[5] In short, these arrays utilize drives that contain redundancy data (i.e., parity data). If one of these drives should fail, all data on the drive – including the redundancy data – must be reconstructed to restore the drive to its former condition and effect repair in accordance with the invention.

For at least the above reasons, the originally-filed disclosure reasonably supports the limitations of claim 1. Accordingly, we will not sustain the examiner's rejection of claims 1-14 under 35 U.S.C. § 112, first paragraph.

We next consider the rejection of claims 16-19 under 35 U.S.C. § 102(e) as being anticipated by Kedem. Anticipation is established only when a single prior art reference discloses, expressly or under the principles of inherency, each

---

8, and 9 recite arrays with drives having parity (redundancy) data (RAID-3 and RAID-5 arrays). Dependent claims 6, 7, and 10-13, however, recite arrays with drives that do not contain parity (redundancy) data (RAID 1 and 1/0 arrays). See RAID Technology, Technick.net, at http://www.technick.net/public/code/cp_dpage.php?aiocp_dp=guide_raid, at 3-4 (last visited Nov. 2, 2006) ("RAID Technology"). See also id., at 7 ("Parity is used by RAID levels 2, 3, 4, and 5. RAID 1 does not use parity because all data is completely duplicated (mirrored)."). The issue is not before us, however, and we leave the question of the scope of the claim to the examiner and the appellant.

[5] See RAID Technology, at 3-4.

and every element of a claimed invention as well as disclosing structure which is capable of performing the recited functional limitations. RCA Corp. v. Applied Digital Data Systems, Inc., 730 F.2d 1440, 1444, 221 USPQ 385, 388 (Fed. Cir. 1984); W.L. Gore and Associates, Inc. v. Garlock, Inc., 721 F.2d 1540, 1554, 220 USPQ 303, 313 (Fed. Cir. 1983).

The examiner has indicated how the claimed invention is deemed to be fully met by the disclosure of Kedem [answer, pages 3-5]. Regarding independent claim 16, appellant argues that Kedem does not disclose reconstructing each data block of the failed disk drive in the redundancy group, and writing each reconstructed data block to the mirrored subsystem as claimed [brief, page 17; emphasis added]. Specifically, appellant contends that "reconstructing each data block" of the failed drive as claimed must be construed as reconstructing all data blocks – both logical and redundancy [brief, page 18]. With this construction, appellant argues that Kedem does not anticipate the claim since Kedem does not reconstruct redundancy data of the failed drive, but rather copies only logical data to temporary drives [id.; reply brief, page 8]. Similarly, appellant contends that Kedem does not anticipate independent claim 18 since Kedem does not reconstruct redundancy data blocks [brief, page 18].

We will sustain the examiner's anticipation rejection. In short, appellant's arguments are not commensurate with the scope of the claims. Although Kedem's mirroring system does not copy the parity data stored on failing device 34 to the spare device 31, each of the failed device's logical volumes are

nevertheless copied to the spare device 31 [Kedem, col. 5, lines 5-17; Figs. 4 and 5]. In our view, Kedem's reconstructing the logical blocks of the failed drive fully meets the limitation calling for "reconstructing each data block" giving the term its broadest reasonable interpretation. We decline to adopt appellant's construction that "each data block" necessarily includes redundancy data blocks. As we noted previously, certain RAID arrays have drives that do not contain parity data (redundancy data), but rather contain only logical data. In short, the scope and breadth of the term "data block" as claimed does not preclude logical data. Accordingly, "reconstructing each data block" as claimed does not preclude Kedem's copying of logical data blocks.

Since Kedem fully meets independent claims 16 and 18, we will sustain the examiner's anticipation rejection of those claims. Since appellant has not separately argued the patentability of dependent claims 17 and 19, these claims fall with independent claims 16 and 18. See In re Nielson, 816 F.2d 1567, 1572, 2 USPQ2d 1525, 1528 (Fed. Cir. 1987). See also 37 CFR § 41.37(c)(vii).

In summary, we have not sustained the examiner's rejection with respect to claims 1-14. We have, however, sustained the examiner's rejection with respect to claims 16-19. Therefore, the decision of the examiner rejecting claims 1-14 and 16-19 is affirmed-in-part.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 CFR § 1.136(a)(1)(iv).

AFFIRMED-IN-PART

Jerry Smith
JERRY SMITH)
Administrative Patent Judge)
)
)
)
) BOARD OF PATENT
LANCE LEONARD BARRY) APPEALS AND
Administrative Patent Judge) INTERFERENCES
)
Allen MacDonald)
)
ALLEN R. MACDONALD)
Administrative Patent Judge)

JS/jaj/rwk

BROMBERG & SUNSTEIN LLP
125 SUMMER STREET
BOSTON, MA 02110-1618

RAID Technology

Reference and Sources:

- The most part of text in this guide has been taken from copyrighted document of Adaptec, Inc. on site (www.adaptec.com)
- Perceptive Solutions, Inc.

RAID stands for **R**edundant **A**rray of **I**nexpensive (or sometimes "Independent") **D**isks.

RAID is a method of combining several hard disk drives into one logical unit (two or more disks grouped together to appear as a single device to the host system). RAID technology was developed to address the fault-tolerance and performance limitations of conventional disk storage. It can offer fault tolerance and higher throughput levels than a single hard drive or group of independent hard drives. While arrays were once considered complex and relatively specialized storage solutions, today they are easy to use and essential for a broad spectrum of client/server applications.

[History] [The driving factors behind RAID] [RAID Levels] [Types Of RAID] [Server Technology Comparison] [Parity] [Fault tolerance]

History

RAID technology was first defined by a group of computer scientists at the University of California at Berkeley in 1987. The scientists studied the possibility of using two or more disks to appear as a single device to the host system.
Although the array's performance was better than that of large, single-disk storage systems, reliability was unacceptably low. To address this, the scientists proposed redundant architectures to provide ways of achieving storage fault tolerance. In addition to defining RAID levels 1 through 5, the scientists also studied data striping -- a non-redundant array configuration that distributes files across multiple disks in an array. Often known as RAID 0, this configuration actually provides no data protection. However, it does offer maximum throughput for some data-intensive applications such as desktop digital video production.

The driving factors behind RAID

A number of factors are responsible for the growing adoption of arrays for critical network storage.

More and more organizations have created enterprise-wide networks to improve productivity and streamline information flow. While the distributed data stored on network servers provides substantial cost benefits, these savings can be quickly offset if information is frequently lost or becomes inaccessible. As today's applications create larger files, network storage needs have increased proportionately. In addition, accelerating CPU speeds have outstripped data transfer rates to storage media, creating bottlenecks in today's systems.

RAID storage solutions overcome these challenges by providing a combination of outstanding data availability, extraordinary and highly scalable performance, high capacity, and recovery with no loss of data or interruption of user access.

By integrating multiple drives into a single array -- which is viewed by the network operating system as a single disk drive -- organizations can create cost-effective, minicomputersized solutions of up to a terabyte or more of storage.

RAID Levels

There are several different RAID "levels" or redundancy schemes, each with inherent cost, performance, and availability (fault-tolerance) characteristics designed to meet different storage needs. No individual RAID level is inherently superior to any other. Each of the five array architectures is well-suited for certain types of applications and computing environments. For client/server applications, storage systems based on RAID levels 1, 0/1, and 5 have been the most widely used. This is because popular NOSs such as Windows NT® Server and NetWare manage data in ways similar to how these RAID architectures perform.

RAID 0 - RAID 1 - RAID 2 - RAID 3 - RAID 4 - RAID 5 - RAID 0/1 (or RAID 10)

RAID 0

Data striping without redundancy (no protection).

- **Minimum number of drives:** 2
- **Strengths:** Highest performance.
- **Weaknesses:** No data protection; One drive fails, all data is lost.

DRIVE 1	DRIVE 2
Data A	Data A
Data B	Data B
Data C	Data C

RAID 1

Disk mirroring.

- **Minimum number of drives:** 2
- **Strengths:** Very high performance; Very high data protection; Very minimal penalty on write performance.
- **Weaknesses:** High redundancy cost overhead; Because all data is duplicated, twice the storage capacity is required.

Mirroring	
Standard Host Adapter	
DRIVE 1	DRIVE 2
Data A	Data A
Data B	Data B
Data C	Data C
Original Data	**Mirrored Data**

Duplexing	
Standard Host Adapter 1	Standard Host Adapter 2
DRIVE 1	DRIVE 2
Data A	Data A
Data B	Data B
Data C	Data C
Original Data	**Mirrored Data**

RAID 2

No practical use.

- **Minimum number of drives:** Not used in LAN
- **Strengths:** Previously used for RAM error environments correction (known as Hamming Code) and in disk drives before he use of embedded error correction.
- **Weaknesses:** No practical use; Same performance can be achieved by RAID 3 at lower cost.

RAID 3

Byte-level data striping with dedicated parity drive.

- **Minimum number of drives:** 3
- **Strengths:** Excellent performance for large, sequential data requests.
- **Weaknesses:** Not well-suited for transaction-oriented network applications; Single parity drive does not support multiple, simultaneous read and write requests.

RAID 4

Block-level data striping with dedicated parity drive.

- **Minimum number of drives:** 3 (Not widely used)
- **Strengths:** Data striping supports multiple simultaneous read requests.
- **Weaknesses:** Write requests suffer from same single parity-drive bottleneck as RAID 3; RAID 5 offers equal data protection and better performance at same cost.

RAID 5

Block-level data striping with distributed parity.

- **Minimum number of drives:** 3
- **Strengths:** Best cost/performance for transaction-oriented networks; Very high performance, very high data protection; Supports multiple simultaneous reads and writes; Can also be optimized for large, sequential requests.
- **Weaknesses:** Write performance is slower than RAID 0 or RAID 1.

DRIVE 1	DRIVE 2	DRIVE 3
Parity A	Data A	Data A
Data B	Parity B	Data B
Data C	Data C	Parity C

RAID 0/1 - RAID 10

Combination of RAID 0 (data striping) and RAID 1 (mirroring).
Note that RAID 10 is another name for RAID (0+1) or RAID 0/1.

- **Minimum number of drives:** 4
- **Strengths:** Highest performance, highest data protection (can tolerate multiple drive failures).
- **Weaknesses:** High redundancy cost overhead; Because all data is duplicated, twice the storage capacity is required; Requires minimum of four drives.

DRIVE 1	DRIVE 2	DRIVE 3	DRIVE 4
Data A	Data A	mA	mA
Data B	Data B	mB	mB
Data C	Data C	mC	mC

Original Data	Original Data	Mirrored Data	Mirrored Data

Types Of RAID

There are three primary array implementations: software-based arrays, bus-based array adapters/controllers, and subsystem-based external array controllers. As with the various RAID levels, no one implementation is clearly better than another -- although software-based arrays are rapidly losing favor as high-performance, low-cost array adapters become increasingly available. Each array solution meets different server and network requirements, depending on the number of users, applications, and storage requirements.

It is important to note that all RAID code is based on software. The difference among the solutions is where that software code is executed -- on the host CPU (software-based arrays) or offloaded to an on-board processor (bus-based and external array controllers).

	Description	Advantages
Software-based RAID	Primarily used with entry-level servers, software-based arrays rely on a standard host adapter and execute all I/O commands and mathematically intensive RAID algorithms in the host server CPU. This can slow system performance by increasing host PCI bus traffic, CPU utilization, and CPU interrupts. Some NOSs such as NetWare and Windows NT include embedded RAID software. The chief advantage of this embedded RAID software has been its lower cost compared to higher-priced RAID alternatives. However, this advantage is disappearing with the advent of lower-cost, bus-based array adapters.	• Low price • Only requires a standard controller.
Hardware-based RAID	Unlike software-based arrays, bus-based array adapters/controllers plug into a host bus slot [typically a 133 MByte (MB)/sec PCI bus] and offload some or all of the I/O commands and RAID operations to one or more secondary processors. Originally used only with mid- to high-end servers due to cost, lower-cost bus-based array adapters are now available specifically for entry-level server network applications. In addition to offering the fault-tolerant benefits of RAID, bus-based array adapters/controllers perform connectivity functions that are similar to	• Data protection and performance benefits of RAID • More robust fault-tolerant features and increased performance versus software-based RAID.

	standard host adapters. By residing directly on a host PCI bus, they provide the highest performance of all array types. Bus-based arrays also deliver more robust fault-tolerant features than embedded NOS RAID software. As newer, high-end technologies such as Fibre Channel become readily available, the performance advantage of bus-based arrays compared to external array controller solutions may diminish.	
External Hardware RAID Card	Intelligent external array controllers "bridge" between one or more server I/O interfaces and single- or multiple-device channels. These controllers feature an on-board microprocessor, which provides high performance and handles functions such as executing RAID software code and supporting data caching. External array controllers offer complete operating system independence, the highest availability, and the ability to scale storage to extraordinarily large capacities (up to a terabyte and beyond). These controllers are usually installed in networks of stand alone Intel-based and UNIX-based servers as well as clustered server environments.	• OS independent • Build super high-capacity storage systems for high-end servers.

Server Technology Comparison

	UDMA	**SCSI**	**Fibre Channel**
Best Suited For	Low-cost entry level server with limited expandability	Low to high-end server when scalability is desired	Server-to-Server campus networks
Advantages	• Uses low-cost ATA drives	• Performance: up to 160 MB/s • Reliability • Connectivity to the largest variety of peripherals • Expandability	• Performance: up to 100 MB/s • Dual active loop data path capability • Infinitely scalable

Parity

The concept behind RAID is relatively simple. The fundamental premise is to be able to recover data on-line in the event of a disk failure by using a form of redundancy called parity. In its simplest form, parity is an addition of all the drives used in an array. Recovery from a drive failure is achieved by reading the remaining good data and checking it against parity data stored by the array. Parity is used by RAID levels 2, 3, 4, and 5. RAID 1 does not use parity because all data is completely duplicated (mirrored). RAID 0, used only to increase performance, offers no data redundancy at all.

```
 A  +  B  +  C  +  D  =  PARITY

 1  +  2  +  3  +  4  =  10
 1  +  2  +  X  +  4  =  10

       7  +  X        =  10
      -7  +           =  -7
     ---------        ----------
             X            3
          MISSING     RECOVERED
           DATA          DATA
```

Fault tolerance

RAID technology does not prevent drive failures. However, RAID does provide insurance against disk drive failures by enabling real-time data recovery without data loss.

The fault tolerance of arrays can also be significantly enhanced by choosing the right storage enclosure. Enclosures that feature redundant, hot-swappable drives, power supplies, and fans can greatly increase storage subsystem uptime based on a number of widely accepted measures:

- **MTDL:**
  Mean **T**ime to **D**ata **L**oss. The average time before the failure of an array component causes data to be lost or corrupted.

- **MTDA:**
  Mean **T**ime between **D**ata **A**ccess (or availability). The average time before non-redundant components fail, causing data inaccessibility without loss or corruption.

- **MTTR:**
  Mean **T**ime **T**o **R**epair. The average time required to bring an array storage subsystem back to full fault tolerance.

- **MTBF:**
  Mean **T**ime **B**etween **F**ailure. Used to measure computer component average reliability/life expectancy. MTBF is not as well-suited for measuring the

reliability of array storage systems as MTDL, MTTR or MTDA (see below) because it does not account for an array's ability to recover from a drive failure. In addition, enhanced enclosure environments used with arrays to increase uptime can further limit the applicability of MTBF ratings for array solutions.

Back to Hardware Guides